BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images, please do not report the images to the Image Problem Mailbox.

	Type	L #	Hits	Search Text	DBs	Time Stamp	Comments
1	BRS	L1	409	instruction adj word SAME generation	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:11	
2	BRS	L2	2	1 and program adj word and pointer adj register	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 08:55	
3	BRS	L3	48	instruction adj word near3 generation	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 08:56	
4	BRS	L4	9	3 and pointer	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 08:56	
5	BRS	L5	1	instruction adj word SAME generation SAME functional adj unit SAME processor SAME program adj sequence	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:13	
6	BRS	L6	7	instruction adj word SAME generation SAME functional adj unit SAME processor	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:20	
7	BRS	L7	6	6 not 5	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:13	

	Type	L #	Hits	Search Text	DBs	Time Stamp	Comments
8	BRS	L8	18	instruction adj word SAME generat? SAME functional adj unit SAME processor	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:40	
9	BRS	L9	14	8 not 7	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:20	
10	BRS	L10	152	read adj pointer adj register	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:30	
11	BRS	L11	96	10 and write adj pointer adj register	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 12:05	
12	BRS	L12	4	11 and instruction adj word	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 12:07	
13	BRS	L13	33504	(instruction adj word anf generat?).ti.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:40	
14	BRS	L14	4	(instruction adj word and generat?).ti.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:43	

	Type	L #	Hits	Search Text	DBs	Time Stamp	Comments
15	BRS	L15	0	(instruction adj word adj generat?).ti.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:43	
16	BRS	L16	410	(instruction adj word).ti.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:43	
17	BRS	L17	20	(instruction adj word and functional adj unit).ti.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:54	
18	IS&R	L18	31293	(341/1-192).CCLS.	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:54	
19	BRS	L19	98	18 and instruction adj word	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:55	
20	BRS	L20	11	18 and instruction adj word and functional adj unit	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 09:55	
21	BRS	L21	58	11 and counter	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 12:06	

	Type	L #	Hits	Search Text	DBs	Time Stamp	Comments
22	BRS	L22	96	11 and (counter WITH write pointer register)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 12:14	
23	BRS	L23	4	22 and instruction adj word	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 12:07	
24	BRS	L24	5	11 and (counter WITH "write pointer register")	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/07/26 12:14	

Terms used
VLIW and **'instruction word generation'** and **pointer** and **register** and **instruction** and **'read pointer register'** and **'write po**

Sort results by relevance

Display results expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search
Try this search in The ACM G

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown Re

1 An efficient resource-constrained global scheduling technique for superscalar and VLIW processors
Soo-Mook Moon, Kemal Ebcioğlu
December 1992 **ACM SIGMICRO Newsletter , Proceedings of the 25th annual international symposium o Microarchitecture**, Volume 23 Issue 1-2
Full text available: pdf(2.05 MB) Additional Information: full citation, references, citings, index terms

Keywords: VLIW, compile-time parallelization, instruction-level parallelism, superscalar

2 Compiler scheduling: Effective instruction scheduling techniques for an interleaved cache clustered VL
Enric Gibert, Jesús Sánchez, Antonio González
November 2002 **Proceedings of the 35th annual ACM/IEEE international symposium on Microarchitectu**
Full text available: pdf(1.13 MB) Publisher Site Additional Information: full citation, abstract, references, citings, index terms

Clustering is a common technique to overcome the wire delay problem incurred by the evolution of technolog distributed architectures, where the register file, the functional units and the data cache are partitioned, are p effective to deal with these constraints and besides they are very scalable. In this paper effective instruction s techniques for a clustered VLIW processor with a word-interleaved cache are proposed. Such scheduling techn (i) loop unro ...

3 Emerging areas: A new look at exploiting data parallelism in embedded systems
Hillery C. Hunter, Jaime H. Moreno
October 2003 **Proceedings of the international conference on Compilers, architectures and synthesis f systems**
Full text available: pdf(322.12 KB) Additional Information: full citation, abstract, references, index terms

This paper describes and evaluates three architectural methods for accomplishing data parallel computation in programmable embedded system. Comparisons are made between the well-studied *Very Long Instruction Wo Single Instruction Multiple Packed Data (SIMpD)* paradigms; the less-common *Single Instruction Multiple Disj (SIMdD)* architecture is described and evaluated. A taxonomy is defined for data-level parallel archi ...

Keywords: DLP, DSP, ILP, SIMD, VLIW, architecture, data-level parallelism, embedded, media, processor, su parallelism, telecommunications

4 Code size minimization and retargetable assembly for custom EPIC and VLIW instruction formats
Shail Aditya, Scott A. Mahlke, B. Ramakrishna Rau
October 2000 **ACM Transactions on Design Automation of Electronic Systems (TODAES)**, Volume 5 Issue 4

h c g e cf c

Full text available: pdf(568.33 KB) Additional Information: full citation, abstract, references, index terms

PICO is a fully automated system for designing the architecture and the microarchitecture of VLIW and EPIC p serious concern with this class of processors, due to their very long instructions, is their code size. One focus to describe a series of code size minimization techniques used within PICO, some of which are applied during design of the instruction format, while others are applied during program assembly. The design of a retargeta

Keywords: EPIC, VLIW, code size minimization, custom templates, design automation, instruction format de compression, retargetable assembly

5 LLVA: A Low-level Virtual Instruction Set Architecture

Vikram Adve, Chris Lattner, Michael Brukman, Anand Shukla, Brian Gaeke

December 2003 **Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitectu**

Full text available: pdf(196.08 KB) Publisher Site Additional Information: full citation, abstract, index terms

A virtual instruction set architecture (V-ISA) implementedvia a processor-specific software translation layerca flexibility to processor designers. Recentexamples such as Crusoe and DAISY, however, haveused existing ha instruction sets as virtual ISAs,which complicates translation and optimization. In fact,there has been little res specific designs for a virtualISA for processors. This paper proposes a novel virtualISA (LLVA) and a translatio implementi ...

6 A VLIW architecture for a trace scheduling compiler

Robert P. Colwell, Robert P. Nix, John J. O'Donnell, David B. Papworth, Paul K. Rodman

October 1987 **Proceedings of the second international conference on Architectual support for program languages and operating systems**, Volume 15 , 22 , 21 Issue 5 , 10 , 4

Full text available: pdf(1.59 MB) Additional Information: full citation, abstract, citings, index terms

Very Long Instruction Word (VLIW) architectures were promised to deliver far more than the factor of two or current architectures achieve from overlapped execution. Using a new type of compiler which compacts ordina code into long instruction words, a VLIW machine was expected to provide from ten to thirty times the perfor conventional machine built of the same implementation technology.Multiflow Computer, Inc., has now built a TRACE™< ...

7 A study on the number of memory ports in multiple instruction issue machines

Soo-Mook Moon, Kemal Ebcioğlu

December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**

Full text available: pdf(1.28 MB) Additional Information: full citation, references, citings

Keywords: ILP, memory disambiguation, memory ports, speculative loads, static scheduling

8 Compilation: Cluster assignment of global values for clustered VLIW processors

Andrei Terechko, Erwan Le Thénaff, Henk Corporaal

October 2003 **Proceedings of the international conference on Compilers, architectures and synthesis f systems**

Full text available: pdf(330.94 KB) Additional Information: full citation, abstract, references, index terms

In this paper high-level language (HLL) variables that are alive in a whole HLL function, across multiple sched termed as global values. Due to their long live ranges and, hence, large impact on the schedule, the global va different compiler optimizations than local values, which span across only one scheduling unit. The instruction clustered ILP processor, which is responsible for cluster assignment of operations and variables, faces a difficu

Keywords: ILP, VLIW, cluster assignment, compiler, instruction scheduler, register allocation

9 MEDEA workshop: Indirect VLIW memory allocation for the ManArray multiprocessor DSP

h c g e cf c

Nikos P. Pitsianis, Gerald G. Pechanek
March 2003 **ACM SIGARCH Computer Architecture News**, Volume 31 Issue 1

Full text available: pdf(623.03 KB) Additional Information: full citation, abstract, references, index terms

The indirect very long instruction word (iVLIW) architecture and its implementation on the BOPS ManArray fam multiprocessor digital signal processors (DSP) provides a scalable alternative to the wide instruction busses u a multiprocessor VLIW DSP. The ManArray processors indirectly access VLIWs from small caches of VLIWs loc processing element. With this work, we present an algorithm to perform 1) iVLIW instruction memory allocati processing el ...

10 Employing register channels for the exploitation of instruction level parallelism
R. Gupta
February 1990 **ACM SIGPLAN Notices , Proceedings of the second ACM SIGPLAN symposium on Principl of parallel programming**, Volume 25 Issue 3

Full text available: pdf(1.14 MB) Additional Information: full citation, abstract, references, citings, index terms

A multiprocessor system capable of exploiting fine-grained parallelism must support efficient synchronization mechanisms. This paper demonstrates the use of shared register channels as the communication mechanism processors in a multiprocessor chip. A register channel is provided with a synchronization bit that is used to e processor succeeds in reading a channel only after the channel has been written to. In contrast to a VLIW ma with channe ...

11 An instruction-level performance analysis of the Multiflow TRACE 14/300
Michael A. Schuette, John P. Shen
September 1991 **Proceedings of the 24th annual international symposium on Microarchitecture**

Full text available: pdf(1.11 MB) Additional Information: full citation, references, citings, index terms

12 Compiler transformations for high-performance computing
David F. Bacon, Susan L. Graham, Oliver J. Sharp
December 1994 **ACM Computing Surveys (CSUR)**, Volume 26 Issue 4

Full text available: pdf(6.32 MB) Additional Information: full citation, abstract, references, citings, index terms, revie

In the last three decades a large number of compiler transformations for optimizing programs have been impl optimizations for uniprocessors reduce the number of instructions executed by the program using transformat the analysis of scalar quantities and data-flow techniques. In contrast, optimizations for high-performance sup vector, and parallel processors maximize parallelism and memory locality with transformations that rely on tr properties o ...

Keywords: compilation, dependence analysis, locality, multiprocessors, optimization, parallelism, superscala vectorization

13 Rationalized three instruction machine
Sachin V. Chitnis, Manoranjan Satpathy, Sundeep Oberoi
March 1995 **ACM SIGPLAN Notices , Papers from the 1995 ACM SIGPLAN workshop on Intermediate representations**, Volume 30 Issue 3

Full text available: pdf(1.01 MB) Additional Information: full citation, abstract, index terms

The declarative nature of functional programming languages causes many difficulties in their efficient implem conventional machines. The problem is much harder when the language has non-strict (lazy) semantics. Abst serve as an intellectual aid in bridging the semantic gap between such languages and the conventional von Ne architecture. However they become more and more complex with time as efficiency considerations force the i the machine to gr ...

Keywords: abstract machines, compiling and optimizations, control flow analysis, functional programming

14 Microprocessor architecture: Increasing the number of effective registers in a low-power processor usin

register file
Rajiv A. Ravindran, Robert M. Senger, Eric D. Marsman, Ganesh S. Dasika, Matthew R. Guthaus, Scott A. Mahlke Brown
October 2003 **Proceedings of the international conference on Compilers, architectures and synthesis f systems**
Full text available: pdf(450.71 KB) Additional Information: full citation, abstract, references, index terms

Low-power embedded processors utilize compact instruction encodings to achieve small code size. Instruction bits are common. Such encodings place tight restrictions on the number of bits available to encode operand s thus on the number of architected registers. The central problem with this approach is that performance and sacrificed as the burden of operand supply is shifted from the register file to the memory due to the limited n register ...

Keywords: embedded processor, graph partitioning, instruction encoding, low-power, register window, windo

15 Dynamically scheduled VLIW processors
B. Ramakrishna Rau
December 1993 **Proceedings of the 26th annual international symposium on Microarchitecture**
Full text available: pdf(1.64 MB) Additional Information: full citation, references, citings

Keywords: VLIW processors, dynamic scheduling, multiple operation issue, out-of-order execution, scoreboa

16 PipeRench implementation of the instruction path coprocessor
Yuan Chou, Pazhani Pillai, Herman Schmit, John Paul Shen
December 2000 **Proceedings of the 33rd annual ACM/IEEE international symposium on Microarchitectu**
Full text available: pdf(336.33 KB) ps(2.57 MB) Publisher Site Additional Information: full citation, references, citings, index terms

17 Instruction path coprocessors
Yuan Chou, John Paul Shen
May 2000 **ACM SIGARCH Computer Architecture News , Proceedings of the 27th annual internation on Computer architecture**, Volume 28 Issue 2
Full text available: pdf(134.64 KB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents the concept of an Instruction Path Coprocessor (I-COP), which is a programmable on-chip with its own mini-instruction set, that operates on the core processor's instructions to transform them into an that can be more efficiently executed. It is located off the critical path of the core processor to ensure that it d negatively impact the core processor's cycle time or pipeline depth. An I-COP is highly versatile and can be us

18 Register connection: a new approach to adding registers into instruction set architectures
Tokuzo Kiyohara, Scott Mahlke, William Chen, Roger Bringmann, Richard Hank, Sadun Anik, Wen-Mei Hwu
May 1993 **ACM SIGARCH Computer Architecture News , Proceedings of the 20th annual internation on Computer architecture**, Volume 21 Issue 2
Full text available: pdf(1.07 MB) Additional Information: full citation, abstract, references, citings, index terms

Code optimization and scheduling for superscalar and superpipelined processors often increase the register re programs. For existing instruction sets with a small to moderate number of registers, this increased register r be a factor that limits the effectivess of the compiler. In this paper, we introduce a new architectural method of extended registers into an architecture. Using a novel concept of connection, this method allows the data s

19 A fine-grained MIMD architecture based upon register channels
Rajiv Gupta
November 1990 **Proceedings of the 23rd annual workshop and symposium on Microprogramming and**

microarchitecture

Full text available: pdf(1.01 MB) Additional Information: full citation, abstract, references, citings

This paper discusses the use of shared register channels as a data exchange mechanism among processors in MIMD system with a load/store architecture. A register channel is provided with a synchronization bit that is u that a processor succeeds in reading a channel only after a value has been written to the channel. The instruc by this load/store architecture allow both registers and register channels to be used as operand sources and r

Keywords: aliasing, channels, fine-grained parallelism, instruction scheduling, multiprocessor system, paralle

20 Tolerating data access latency with register preloading

William Y. Chen, Scott A. Mahlke, Wen-mei W. Hwu, Tokuzo Kiyohara, Pohua P. Chang

August 1992 **Proceedings of the 6th international conference on Supercomputing**

Full text available: pdf(970.85 KB) Additional Information: full citation, abstract, references, citings, index terms

By exploiting fine grain parallelism, superscalar processors can potentially increase the performance of future supercomputers. However, supercomputers typically have a long access delay to their first level memory whic restrict the performance of superscalar processors. Compilers attempt to move load instructions far enough a latency. However, conventional movement of load instructions is limited by data dependence analysis. This pa simple har ...

Keywords: VLIW/superscalar processor, data dependence analysis, load latency, register file, register preloa